

1 Claims

2 Having thus described our invention, what we claim
3 as new and desire to secure by Letters Patent is as
4 follows:

5 1. A method for processing network packets at an
6 intermediate node, the method comprising:

7 forming a first connection between a first node and
8 the intermediate node this first connection including:

9 a first flow originating at a first source
10 flow end point on the first node and terminating at
11 a first destination flow end point on the
12 intermediate node, wherein processing at the first
13 node associated with the first source flow end point
14 conforms to a first protocol, and

15 a second flow originating at a second source
16 flow end point on the intermediate node and
17 terminating at a second destination flow end point
18 on the first node, wherein processing at the first
19 node associated with the second destination flow end
20 point conforms to a second protocol;

21 forming a second connection between a second node
22 and the intermediate node this second connection
23 including:

24 a third flow originating at a third source
25 flow end point on the intermediate node and

1 terminating at a third destination flow end point on
2 the second node, wherein processing at the second
3 node associated with the third destination flow end
4 point conforms to a third protocol, and

5 a fourth flow originating at a fourth source
6 flow end point on the second node and terminating at
7 a fourth destination flow end point on the
8 intermediate node, wherein processing at the second
9 node associated with the fourth source flow end
10 point conforms to a fourth protocol,

11 such that a given flow originates at a source flow
12 end point on a source node and terminates at a
13 destination flow end point on a destination node and
14 data written to the source flow end point of a given
15 flow can subsequently be read from the destination
16 flow end point of the given flow without traversing
17 any intervening flow end points, and

18 splicing the first flow and third flow to form a
19 first composite flow originating at the first source
20 flow end point on the first node and terminating at the
21 third destination flow end point on the second node in
22 a manner whereby the second flow and the fourth flow
23 remain unchanged.

24 2. A method as recited in claim 1, wherein the
25 step of splicing the first flow and third flow to form
26 the first composite flow allows

1 processing at the first node associated with the
2 first source flow end point to remain unmodified and
3 continue to conform to the first protocol,
4 processing at the first node associated with the
5 second destination flow end point to remain unmodified
6 and continue to conform to the second protocol,
7 processing at the second node associated with the
8 third destination flow end point to remain unmodified
9 and continue to conform to the third protocol, and
10 processing at the second node associated with the
11 fourth source flow end point to remain unmodified and
12 continue to conform to the fourth protocol.

13 3. A method recited in claim 1, wherein the first
14 node and the second node are the same node.

15 4. A method recited in claim 1, wherein the first
16 node and intermediate node are the same node.

17 5. A method recited in claim 1, wherein the second
18 node and the intermediate node are the same node.

19 6. A method recited in claim 1, wherein the first
20 node, the second node, and the intermediate node are
21 all the same node.

22 7. A method recited in claim 1, wherein the first
23 protocol and the second protocol are the same protocol.

24 8. A method recited in claim 1, wherein the third
25 protocol and the fourth protocol are the same protocol.

1 9. A method recited in claim 1, wherein the first
2 protocol, the second protocol, the third protocol, and
3 the fourth protocol are all the same protocol.

4 10. A method recited in claim 1, wherein the step of
5 splicing the first flow and third flow to form the
6 first composite flow includes:

7 identifying a first set of packets received from the
8 first node including all packets containing information
9 pertaining to the first source flow end point and all
10 packets containing information pertaining to the second
11 destination flow end point and performing the following
12 four steps (a), (b), (c) and (d) on each packet in
13 this first set of packets;

14 (a) processing any information in the packet
15 pertaining to the second flow according to the
16 second protocol;

17 (b) replacing any information in the packet
18 pertaining to the second flow with the corresponding
19 information pertaining to the fourth flow;

20 (c) modifying the packet so that any information in
21 the packet pertaining to the flow from the first
22 source flow end point will appear to the second node
23 as pertaining to the flow to the third destination
24 flow end point thus establishing a correspondence
25 between data received by the intermediate node from
26 the first source flow end point and data sent by the

1 intermediate node to the third destination flow end
2 point, and

3 (d) sending each packet so processed to the second
4 node;

5 identifying a second set of packets received from
6 the second node including all packets containing
7 information pertaining to the third destination flow
8 end point and all packets containing information
9 pertaining to the fourth source flow end point and
10 performing the following steps (d), (e), and (f), on
11 each packet in this second set of packets;

12 (d) identifying any information in the packet
13 pertaining to the fourth flow and processing such
14 information according to the fourth protocol;

15 (e) modifying any information in the packet
16 pertaining to the flow to the third destination flow
17 end point so the information instead pertains to the
18 flow from the first source flow end point according to
19 the correspondence between data received by the
20 intermediate node from the first source flow end point
21 and data sent by the intermediate node to the third
22 destination flow end point established in step (b);

23 (f) sending to the first node zero or more packets
24 containing any information resulting from step (e).

1 11. A method recited in claim 10, wherein step (c)
2 includes determining if the packet to be sent should be
3 fragmented and fragmenting it if so.

4 12. A method recited in claim 10, further comprising
5 determining if each packet in the first set of packets
6 contains data written to the first source flow end
7 point and nullifying steps (b), (c) and (d) if not.

8 13. A method recited in claim 10, further comprising
9 determining if each packet in the second set of packets
10 contains data written to the fourth source flow end
11 point, determining if each packet in the second set of
12 packets contains no information pertaining to the flow
13 to the third destination flow end point that has not
14 already been processed by a previous application of
15 step (e), and nullifying steps (e) and (f) for a given
16 packet if both determinations are true for the given
17 packet.

18 14. A method recited in claim 1, wherein the first
19 protocol and the third protocol each associate a
20 sequence number with each byte, packet or other unit of
21 data sent across a flow further comprising the step of
22 maintaining a one to one mapping between sequence
23 numbers associated by the first protocol with each
24 byte, packet, or other unit of data received by the
25 intermediate node from the first source flow end point
26 and sequence numbers associated by the second protocol
27 with each byte, packet, or other unit of data sent by
28 the intermediate node to the third destination flow end
29 point.

1 15. A method recited in claim 1, further comprising
2 the step of sending data stored in buffers associated
3 with the third source flow end point to the third
4 destination flow end point according to the third
5 protocol.

6 16. A method recited in claim 1, further comprising
7 the step of sending data stored in buffers associated
8 with the first destination flow end point to the third
9 destination flow end point according to the third
10 protocol.

11 17. A method recited in claim 1, further comprising
12 the step of determining if the first node has shut down
13 the flow originating at the first source flow end point
14 and shutting down the flow terminating at the third
15 destination flow end point if this determination is
16 true.

17 18. A method recited in claim 1, further comprising
18 determining if the first node has shut down the flow
19 originating at the first source flow end point and
20 eliminating the first composite flow and recreating the
21 third flow if this determination is true.

22 19. A method recited in claim 1, further comprising
23 the step of making a copy of data received from the
24 first node by the intermediate node such that the copy
25 may subsequently be read at the intermediate node.

1 20. A method recited in claim 1, further comprising
2 the steps of:

3 reading an amount of data from the fourth
4 destination flow end point;

5 storing the amount of data in a format satisfying a
6 need from the group of needs consisting of:

7 reducing the number of bits required to
8 store the data,

9 encrypting the data, reducing the number or
10 amount of resources required to transmit the data,

11 reducing the number or amount of resources
12 required to display the data, and

13 any combination of these needs; and

14 sending the stored data to the first node.

15 21. An apparatus comprising:

16 a first node, a second node, an intermediate node;

17 a first processor that forms a first connection
18 between a first node and the intermediate node the
19 first connection including:

20 a first flow originating at a first source
21 flow end point on the first node and terminating at
22 a first destination flow end point on the
23 intermediate node, wherein processing at the first
24 node associated with the first source flow end point
25 conforms to a first protocol, and

1 a second flow originating at a second source
2 flow end point on the intermediate node and
3 terminating at a second destination flow end point
4 on the first node, wherein processing at the first
5 node associated with the second destination flow end
6 point conforms to a second protocol;

7 a second processor that forms a second connection
8 between a second node and the intermediate node the
9 second connection including:

10 a third flow originating at a third source
11 flow end point on the intermediate node and
12 terminating at a third destination flow end point on
13 the second node, wherein processing at the second
14 node associated with the third destination flow end
15 point conforms to a third protocol, and

16 a fourth flow originating at a fourth source
17 flow end point on the second node and terminating at
18 a fourth destination flow end point on the
19 intermediate node, wherein processing at the second
20 node associated with the fourth source flow end
21 point conforms to a fourth protocol,

22 such that a given flow originates at a source flow
23 end point on a source node and terminates at a
24 destination flow end point on a destination node and
25 data written to the source flow end point of a given
26 flow can subsequently be read from the destination

1 flow end point of the given flow without traversing
2 any intervening flow end points, and

3 a third processor that splices the first flow and
4 third flow to form a first composite flow originating
5 at the first source flow end point on the first node
6 and terminating at the third destination flow end point
7 on the second node in a manner whereby the second flow
8 and the fourth flow remain unchanged.

9 22. A method recited in claim 1, further comprising:

10 forming a third connection between a third node and
11 the intermediate node the third connection including:

12 a fifth flow originating at a fifth source flow end
13 point on the intermediate node and terminating at a
14 fifth destination flow end point on the third node,
15 wherein processing at the third node associated with
16 the fifth destination flow end point conforms to a
17 fifth protocol, and

18 a sixth flow originating at a sixth source flow end
19 point on the third node and terminating at a sixth
20 destination flow end point on the intermediate node,
21 wherein processing at the sixth node associated with
22 the fourth source flow end point conforms to a sixth
23 protocol,

24 splicing the sixth flow and second flow to form a
25 third composite flow originating at the sixth source
26 flow end point on the third node and terminating at the

1 second destination flow end point on the first node in
2 a manner whereby the fourth flow and the fifth flow
3 remain unchanged.

4 23. A method recited in claim 1, further comprising
5 the step of reading data from the first destination
6 flow end point prior to the step of splicing.

7 24. A method recited in claim 1, further comprising
8 the step of writing data to the third source flow end
9 point prior to the step of splicing.

10 25. A method recited in claim 1, further comprising
11 the step of determining if the step of splicing should
12 be performed based on a criterion selected from the
13 group of criteria consisting of:

14 data read from the first destination flow end point,
15 data read from the fourth destination flow end
16 point,

17 data written to the second source flow end point,

18 data written to the third source flow end point,

19 an address associated with the first node,

20 an address associated with the second node,

21 an address associated with the intermediate node,

22 an address associated with the first flow,

23 an address associated with the second flow,

24 an address associated with the third flow,

25 an address associated with the fourth flow,

26 an estimate of the available bandwidth along the
27 first flow,

28 an estimate of the available bandwidth along the
29 second flow,

1 an estimate of the available bandwidth along the
2 third flow,
3 an estimate of the available bandwidth along the
4 fourth flow,
5 an estimate of the end to end latency associated
6 with the first flow,
7 an estimate of the end to end latency associated
8 with the second flow,
9 an estimate of the end to end latency associated
10 with the third flow,
11 an estimate of the end to end latency associated
12 with the fourth flow,
13 a receive window size advertised by the first node,
14 a receive window size advertised by the second node,
15 a receive window size advertised by the intermediate
16 node,
17 a number of network hops between the first node and
18 the intermediate node,
19 a number of network hops between the intermediate
20 node and the second node,
21 a number of network hops between the first node and
22 the second node,
23 a protocol associated with the first flow,
24 a protocol associated with the second flow,
25 a protocol associated with the third flow,
26 a protocol associated with the fourth flow,
27 an estimate of the amount of data that will be
28 written to the first source flow end point,
29 an estimate of the amount of data that will be
30 written to the second source flow end point,
31 an estimate of the amount of data that will be
32 written to the third source flow end point,

1 an estimate of the amount of data that will be
2 written to the fourth source flow end point,
3 configuration information stored at or accessible
4 from the intermediate node,
5 information received from the first node pertaining
6 to quality of service,
7 information received from the second node pertaining
8 to quality of service , and
9 any combination of these criteria.

10 26. A method recited in claim 1, further comprising
11 the step of determining the network addresses with
12 which the first flow, second flow, third flow, and
13 fourth flow are established based on a criterion
14 selected from the group of criteria consisting of:
15 data read from the first destination flow end point,
16 data read from the fourth destination flow end
17 point,
18 data written to the second source flow end point,
19 data written to the third source flow end point,
20 a network address associated with a potential or
21 actual first node,
22 a network address associated with a potential or
23 actual second node,
24 a network address associated with the intermediate
25 node,
26 an address associated with the first flow,
27 an address associated with the second flow,
28 an address associated with the third flow,
29 an address associated with the fourth flow,
30 an estimate or prediction of the available bandwidth
31 along the first flow

1 an estimate or prediction of the available bandwidth
2 along the second flow,
3 an estimate or prediction of the available bandwidth
4 along the third flow
5 an estimate or prediction of the available bandwidth
6 along the fourth flow,
7 an estimate or prediction of the end to end latency
8 along the first flow,
9 an estimate or prediction of the end to end latency
10 along the second flow,
11 an estimate or prediction of the end to end latency
12 along the third flow,
13 an estimate or predication of the end to end latency
14 along the fourth flow,
15 a receive window size advertised by the first node,
16 a receive window size advertised by the second node,
17 a receive window size advertised by the intermediate
18 node,
19 a number of network hops between the first node and
20 the intermediate node,
21 a number of network hops between the intermediate
22 node and the second node,
23 a number of hops between the first node and the
24 second node,
25 a protocol associated with the first flow,
26 A protocol associated with the second flow,
27 an estimate of the amount of data that will be
28 written to the first source flow end point,
29 an estimate of the amount of data that will be
30 written to the second source flow end point,
31 an estimate of the amount of data that will be
32 written to the third source flow end point,

1 an estimate of the amount of data that will be
2 written to the fourth source flow end point,
3 configuration information stored at or accessible
4 from the intermediate node,
5 information received from the first node pertaining
6 to quality of service,
7 information received from the second node pertaining
8 to quality of service, and
9 any combination of these criteria.

10 27. A method recited in claim 10, wherein each of
11 the first protocol, the second protocol, the third
12 protocol and the fourth protocol are Transmission
13 Control Protocol (TCP) in conjunction with either
14 version 4 or version 6 of Internet Protocol (IP) and
15 wherein step (c) of claim 1, includes the following
16 steps (a), (b), (c), (d), and (e):

17 (a) setting the source IP address in the IP header
18 to the local IP address associated with the third
19 source flow end point;
20 (b) setting the destination IP address in the IP
21 header to the remote IP address associated with the
22 third source flow end point;
23 (c) setting the source port number in the TCP header
24 to the local port number associated with the third
25 source flow end point;
26 (d) setting the destination port number in the TCP
27 header to the remote port number associated with the
28 third source flow end point, and
29 (e) modifying the sequence (SEQ) number in the TCP
30 header;

1 and wherein step (b) of claim 10, includes the
2 following steps (f), (g), (h), and (i):

3 (f) replacing the acknowledgment (ACK) number in the
4 TCP header;

5 (g) replacing the window value in the TCP header;

6 (h) modifying or recalculating the TCP checksum in
7 the TCP header, and

8 (i) modifying or recalculating the IP checksum in
9 the IP header;

10 and wherein step (e) of claim 1, includes the
11 following steps (j), (k), (l), (m), (n), (o), (p) and
12 (q):

13 (j) setting the source IP address in the IP header
14 to the local IP address associated with the first
15 destination flow end point;

16 (k) setting the destination IP address in the IP
17 header to the remote IP address associated with the
18 first destination flow end point;

19 (l) setting the source port number in the TCP header
20 to the local port number associated with the first
21 destination flow end point;

22 (m) setting the destination port number in the TCP
23 header to the remote port number associated with the
24 first destination flow end point;

25 (n) replacing the sequence (SEQ) number in the TCP
26 header;

27 (o) modifying the acknowledgment (ACK) number in the
28 TCP header;

1 (p) modifying the TCP checksum in the TCP header,
2 and
3 (q) modifying the IP checksum in the IP header.

4 28. A method recited in claim 27, wherein any of the
5 checksum calculations performed in steps (h), (i), (p),
6 and (q) is calculated by adjusting the original
7 checksum to account for changes made to the packet at
8 the intermediate node.

9 29. A method recited in claim 27, wherein the first
10 connection extends between a TCP server on the first
11 node and a SOCKS server on the intermediate node and
12 the second connection extends between the SOCKS server
13 on the intermediate node and a SOCKS client on the
14 second node further comprising parsing data received
15 from either the SOCKS client or the TCP server.

16 30. A method recited in claim 27, wherein the first
17 connection extends between a SOCKS client on the first
18 node and a SOCKS server on the intermediate node and
19 the second connection extends between the SOCKS server
20 on the intermediate node and a TCP server on the second
21 node further comprising parsing data received from
22 either the SOCKS client or the TCP server.

23 31. A method recited in claim 27, wherein the first
24 connection extends between a Sun remote procedure call
25 (Sun RPC) server on the first node and a proxy on the
26 intermediate node and the second connection extends
27 between the proxy on the intermediate node and a Sun
28 RPC client on the second node further comprising

1 parsing data received from either the Sun RPC client or
2 the Sun RPC server.

3 32. A method recited in claim 27, wherein the first
4 connection extends between a Sun remote procedure call
5 (Sun RPC) client on the first node and a proxy on the
6 intermediate node and the second connection extends
7 between the proxy on the intermediate node and a Sun
8 RPC server on the second node further comprising
9 parsing data received from either the Sun RPC client or
10 the Sun RPC server.

11 33. A method recited in claim 27, wherein the first
12 connection extends between a hypertext transfer
13 protocol (HTTP) server on the first node and a proxy on
14 the intermediate node and the second connection extends
15 between the proxy on the intermediate node and an HTTP
16 client on the second node further comprising parsing
17 data received from either the HTTP client or the HTTP
18 server.

19 34. A method recited in claim 27, wherein the first
20 connection extends between a hypertext transfer
21 protocol (HTTP) client on the first node and a proxy on
22 the intermediate node and the second connection extends
23 between the proxy on the intermediate node and an HTTP
24 server on the second node further comprising parsing
25 data received from either the HTTP client or the HTTP
26 server.

27 35. A method recited in claim 27, wherein the first
28 connection extends between a file transfer protocol

1 (FTP) server on the first node and a proxy on the
2 intermediate node and the second connection extends
3 between the proxy on the intermediate node and an FTP
4 client on the second node further comprising parsing
5 data received from either the FTP client or the FTP
6 server.

7 36. A method recited in claim 27, wherein the first
8 connection extends between a file transfer protocol
9 (FTP) client on the first node and a proxy on the
10 intermediate node and the second connection extends
11 between the proxy on the intermediate node and an FTP
12 server on the second node further comprising parsing
13 data received from either the FTP client or the FTP
14 server.

15 37. A method recited in claim 27, wherein the first
16 connection extends between a telnet server on the first
17 node and a proxy on the intermediate node and the
18 second connection extends between the proxy on the
19 intermediate node and an telnet client on the second
20 node further comprising parsing data received from
21 either the telnet client or the telnet server.

22 38. A method recited in claim 27, wherein the first
23 connection extends between a telnet client on the first
24 node and a proxy on the intermediate node and the
25 second connection extends between the proxy on the
26 intermediate node and an telnet server on the second
27 node further comprising parsing data received from
28 either the telnet client or the telnet server.

1 39. A method recited in claim 27, wherein the first
2 connection extends between a network news transfer
3 protocol (NNTP) server on the first node and a proxy on
4 the intermediate node and the second connection extends
5 between the proxy on the intermediate node and an NNTP
6 client on the second node further comprising parsing
7 data received from either the NNTP client or the NNTP
8 server.

9 40. A method recited in claim 27, wherein the first
10 connection extends between a network news transfer
11 protocol (NNTP) client on the first node and a proxy on
12 the intermediate node and the second connection extends
13 between the proxy on the intermediate node and an NNTP
14 server on the second node further comprising parsing
15 data received from either the NNTP client or the NNTP
16 server.

17 41. A method recited in claim 27, wherein the first
18 connection extends between a secure shell (SSH) server
19 on the first node and a proxy on the intermediate node
20 and the second connection extends between the proxy on
21 the intermediate node and an SSH client on the second
22 node further comprising parsing data received from
23 either the SSH client or the SSH server.

24 42. A method recited in claim 27, wherein the first
25 connection extends between a secure shell (SSH) client
26 on the first node and a proxy on the intermediate node
27 and the second connection extends between the proxy on
28 the intermediate node and an SSH server on the second

1 node further comprising parsing data received from
2 either the SSH client or the SSH server.

3 43. A method recited in claim 27, wherein the first
4 connection extends between a remote shell (RSH) server
5 on the first node and a proxy on the intermediate node
6 and the second connection extends between the proxy on
7 the intermediate node and an RSH client on the second
8 node further comprising parsing data received from
9 either the RSH client or the RSH server.

10 44. A method recited in claim 27, wherein the first
11 connection extends between a remote shell (RSH) client
12 on the first node and a proxy on the intermediate node
13 and the second connection extends between the proxy on
14 the intermediate node and an RSH server on the second
15 node further comprising parsing data received from
16 either the RSH client or the RSH server.

17 45. A method recited in claim 27, wherein the first
18 connection extends between a simple mail transfer
19 protocol (SMTP) server on the first node and a proxy on
20 the intermediate node and the second connection extends
21 between the proxy on the intermediate node and an SMTP
22 client on the second node further comprising parsing
23 data received from either the SMTP client or the SMTP
24 server.

25 46. A method recited in claim 27, wherein the first
26 connection extends between a simple mail transfer
27 protocol (SMTP) client on the first node and a proxy on
28 the intermediate node and the second connection extends

1 between the proxy on the intermediate node and an SMTP
2 server on the second node further comprising parsing
3 data received from either the SMTP client or the SMTP
4 server.

5 47. A method recited in claim 27, wherein the first
6 connection extends between a post office protocol (POP)
7 server on the first node and a proxy on the
8 intermediate node and the second connection extends
9 between the proxy on the intermediate node and a POP
10 client on the second node further comprising parsing
11 data received from either the POP client or the POP
12 server.

13 48. A method recited in claim 27, wherein the first
14 connection extends between a post office protocol (POP)
15 client on the first node and a proxy on the
16 intermediate node and the second connection extends
17 between the proxy on the intermediate node and a POP
18 server on the second node further comprising parsing
19 data received from either the POP client or the POP
20 server.

21 49. A method recited in claim 27, wherein the first
22 connection extends between a server sending streaming
23 audio and or video on the first node and a proxy on the
24 intermediate node and the second connection extends
25 between the proxy on the intermediate node and a client
26 receiving streaming audio or video on the second node
27 further comprising parsing data received from either
28 the client or the server.

1 50. A method recited in claim 37 wherein the first
2 connection extends between a client reading streaming
3 audio and/or video on the first node and a proxy on the
4 intermediate node and the second connection extends
5 between the proxy on the intermediate node and a server
6 sending streaming audio or video on the second node
7 further comprising parsing data received from either
8 the client or the server.

9 51. A method recited in claim 1, further comprising:

10 forming a third connection between a third node and
11 the intermediate node the third connection including:

12 a fifth flow originating at a fifth source flow end
13 point on the intermediate node and terminating at a
14 fifth destination flow end point on the third node,
15 wherein processing at the third node associated with
16 the fifth destination flow end point conforms to a
17 fifth protocol, and

18 a sixth flow originating at a sixth source flow end
19 point on the third node and terminating at a sixth
20 destination flow end point on the intermediate node,
21 wherein processing at the sixth node associated with
22 the fourth source flow end point conforms to a sixth
23 protocol,

24 splicing the fourth flow and fifth flow to form a
25 second composite flow originating at the fourth source
26 flow end point on the second node and terminating at
27 the fifth destination flow end point on the third node

1 in a manner whereby the second flow and the sixth flow
2 remain unchanged.

3 52. An article of manufacture comprising a computer
4 usable medium having computer readable program code
5 means embodied therein for causing the processing of
6 network packets at an intermediate node, the computer
7 readable program code means in said article of
8 manufacture comprising computer readable program code
9 means for causing a computer to effect:

10 forming a first connection between a first node and
11 the intermediate node this first connection including:

12 a first flow originating at a first source
13 flow end point on the first node and terminating at
14 a first destination flow end point on the
15 intermediate node, wherein processing at the first
16 node associated with the first source flow end point
17 conforms to a first protocol, and

18 a second flow originating at a second source
19 flow end point on the intermediate node and
20 terminating at a second destination flow end point
21 on the first node, wherein processing at the first
22 node associated with the second destination flow end
23 point conforms to a second protocol;

24 forming a second connection between a second node
25 and the intermediate node this second connection
26 including:

1 a third flow originating at a third source
2 flow end point on the intermediate node and
3 terminating at a third destination flow end point on
4 the second node, wherein processing at the second
5 node associated with the third destination flow end
6 point conforms to a third protocol, and

7 a fourth flow originating at a fourth source
8 flow end point on the second node and terminating at
9 a fourth destination flow end point on the
10 intermediate node, wherein processing at the second
11 node associated with the fourth source flow end
12 point conforms to a fourth protocol,

13 such that a given flow originates at a source flow
14 end point on a source node and terminates at a
15 destination flow end point on a destination node and
16 data written to the source flow end point of a given
17 flow can subsequently be read from the destination
18 flow end point of the given flow without traversing
19 any intervening flow end points, and

20 splicing the first flow and third flow to form a
21 first composite flow originating at the first source
22 flow end point on the first node and terminating at the
23 third destination flow end point on the second node in
24 a manner whereby the second flow and the fourth flow
25 remain unchanged.

26 53. An article of manufacture as recited in claim
27 52, wherein the step of splicing the first flow and (
28 third flow to form the first composite flow allows:

1 processing at the first node associated with the
2 first source flow end point to remain unmodified and
3 continue to conform to the first protocol,
4 processing at the first node associated with the
5 second destination flow end point to remain unmodified
6 and continue to conform to the second protocol,
7 processing at the second node associated with the
8 third destination flow end point to remain unmodified
9 and continue to conform to the third protocol, and
10 processing at the second node associated with the
11 fourth source flow end point to remain unmodified and
12 continue to conform to the fourth protocol.

13) 54. An article of manufacture as recited in claim
14 52, wherein the first node and the second node are the
15 same node.

16 55. An article of manufacture as recited in claim
17 52, wherein the first protocol and the third protocol
18 each associate a sequence number with each byte, packet
19 or other unit of data sent across a flow, the computer
20 readable program code means in said article of
21 manufacture further comprising computer readable
22 program code means for causing a computer to effect
23 maintaining a one to one mapping between sequence
24 numbers associated by the first protocol with each
25 byte, packet, or other unit of data received by the
26 intermediate node from the first source flow end point
27 and sequence numbers associated by the second protocol
28 with each byte, packet, or other unit of data sent by
29 the intermediate node to the third destination flow end
30 point.

1 56. An article of manufacture as recited in claim
2 52, wherein the step of splicing the first flow and
3 third flow to form the first composite flow includes:

4 identifying a first set of packets received from the
5 first node including all packets containing information
6 pertaining to the first source flow end point and all
7 packets containing information pertaining to the second
8 destination flow end point and performing the following
9 four steps (a), (b), (c) and (d) on each packet in
10 this first set of packets;

11 (a) processing any information in the packet
12 pertaining to the second flow according to the
13 second protocol;

14 (b) replacing any information in the packet
15 pertaining to the second flow with the corresponding
16 information pertaining to the fourth flow;

17 (c) modifying the packet so that any information in
18 the packet pertaining to the flow from the first
19 source flow end point will appear to the second node
20 as pertaining to the flow to the third destination
21 flow end point thus establishing a correspondence
22 between data received by the intermediate node from
23 the first source flow end point and data sent by the
24 intermediate node to the third destination flow end
25 point, and

1 (d) sending each packet so processed to the second
2 node;

3 identifying a second set of packets received from
4 the second node including all packets containing
5 information pertaining to the third destination flow
6 end point and all packets containing information
7 pertaining to the fourth source flow end point and
8 performing the following steps (d), (e), and (f), on
9 each packet in this second set of packets;

10 (d) identifying any information in the packet
11 pertaining to the fourth flow and processing such
12 information according to the fourth protocol;

13 (e) modifying any information in the packet
14 pertaining to the flow to the third destination flow
15 end point so the information instead pertains to the
16 flow from the first source flow end point according to
17 the correspondence between data received by the
18 intermediate node from the first source flow end point
19 and data sent by the intermediate node to the third
20 destination flow end point established in step (b);

21 (f) sending to the first node zero or more packets
22 containing any information resulting from step (e).

23 57. An article of manufacture as recited in claim
24 56, wherein each of the first protocol, the second
25 protocol, the third protocol and the fourth protocol
26 are Transmission Control Protocol (TCP) in conjunction
27 with either version 4 or version 6 of Internet Protocol

1 (IP) and wherein step (c) of claim 1, includes the
2 following steps (a), (b), (c), (d), and (e):

3 (a) setting the source IP address in the IP header
4 to the local IP address associated with the third
5 source flow end point;

6 (b) setting the destination IP address in the IP
7 header to the remote IP address associated with the
8 third source flow end point;

9 (c) setting the source port number in the TCP header
10 to the local port number associated with the third
11 source flow end point;

12 (d) setting the destination port number in the TCP
13 header to the remote port number associated with the
14 third source flow end point, and

15 (e) modifying the sequence (SEQ) number in the TCP
16 header;

17 and wherein step (b) of claim 10, includes the
18 following steps (f), (g), (h), and (i):

19 (f) replacing the acknowledgment (ACK) number in the
20 TCP header;

21 (g) replacing the window value in the TCP header;

22 (h) modifying or recalculating the TCP checksum in
23 the TCP header, and

24 (i) modifying or recalculating the IP checksum in
25 the IP header;

26 and wherein step (e) of claim 1, includes the
27 following steps (j), (k), (l), (m), (n), (o), (p) and
28 (q):

- 1 (j) setting the source IP address in the IP header
2 to the local IP address associated with the first
3 destination flow end point;
4 (k) setting the destination IP address in the IP
5 header to the remote IP address associated with the
6 first destination flow end point;
7 (l) setting the source port number in the TCP header
8 to the local port number associated with the first
9 destination flow end point;
10 (m) setting the destination port number in the TCP
11 header to the remote port number associated with the
12 first destination flow end point;
13 (n) replacing the sequence (SEQ) number in the TCP
14 header;
15 (o) modifying the acknowledgment (ACK) number in the
16 TCP header;
17 (p) modifying the TCP checksum in the TCP header,
18 and
19 (q) modifying the IP checksum in the IP header.

20 58. A computer program product comprising a
21 computer usable medium having computer readable program
22 code means embodied therein for causing the processing
23 of network packets at an intermediate node, the
24 computer readable program code means in said computer
25 program product comprising computer readable program
26 code means for causing a computer to effect:

27 forming a first connection between a first node and
28 the intermediate node this first connection including:

1 a first flow originating at a first source
2 flow end point on the first node and terminating at
3 a first destination flow end point on the
4 intermediate node, wherein processing at the first
5 node associated with the first source flow end point
6 conforms to a first protocol, and

7 a second flow originating at a second source
8 flow end point on the intermediate node and
9 terminating at a second destination flow end point
10 on the first node, wherein processing at the first
11 node associated with the second destination flow end
12 point conforms to a second protocol;

13 forming a second connection between a second node
14 and the intermediate node this second connection
15 including:

16 a third flow originating at a third source
17 flow end point on the intermediate node and
18 terminating at a third destination flow end point on
19 the second node, wherein processing at the second
20 node associated with the third destination flow end
21 point conforms to a third protocol, and

22 a fourth flow originating at a fourth source
23 flow end point on the second node and terminating at
24 a fourth destination flow end point on the
25 intermediate node, wherein processing at the second
26 node associated with the fourth source flow end
27 point conforms to a fourth protocol,

1 such that a given flow originates at a source flow
2 end point on a source node and terminates at a
3 destination flow end point on a destination node and
4 data written to the source flow end point of a given
5 flow can subsequently be read from the destination
6 flow end point of the given flow without traversing
7 any intervening flow end points, and

8 splicing the first flow and third flow to form a
9 first composite flow originating at the first source
10 flow end point on the first node and terminating at the
11 third destination flow end point on the second node in
12 a manner whereby the second flow and the fourth flow
13 remain unchanged.

1
2
3
4
5
6
7
8
9
10
11
12
13